# IPv6 Security

**Gene Cronk - NSA-IAM, MCP, Sec+, iNet+, Net+ (gcronk@trsg.net)**
**Hackery SME -- The North American IPv6 Task Force**
**Systems Administrator -- The Robin Shepherd Group**

When discussing IPv6 security, three major subjects tend to rear their heads (some uglier than others). The first subject is native IPv6. While not being a true security issue in itself, all protocol implementations have security flaws. All protocol implementations are designed by man, and therefore, fundamentally flawed. IPv6 is no exception to this rule; however, IPv6 was built to address the 30 years of limitations that the explosion of the IPv4 protocol made apparent. The IPv4 protocol was not originally designed for the stress tests upon which we currently base our entire infrastructure.

The second major issue with native IPv6 currently is a lack of support. At this point, we are confronted with the issue of the chicken and the egg. The protocol isn't widely deployed, therefore there are few applications available for it. By the same token, there are few applications written for IPv6, so it is not widely deployed. Many of the major players in the IT industry are trying to change this. Microsoft has included (albeit limited) support for IPv6 in their latest flagship operating systems. Cisco, whom by many is considered the routing backbone of the Internet, has been including IPv6 support in many of their latest IOSes. The Open Source community has been leading the pack with the KAME project for the BSD based operating systems, as well as the USAGI project for Linux kernels. While these are great leaps towards the goal of global adoption of IPv6, they are simply not enough.

A caveat of not having application support for IPv6 is that there is really no way to perform effective security assessments of one's own network. There are a few tools available, but not enough to do a full assessment of possible vulnerabilities, even with the limited arsenal of IPv6 capable F/OSS and COTS security assessment software. Among the tools that do exist are:

- **NMAP** (http://www.insecure.org) – An open source port scanner with limited (TCP SYN scan) support.
- **HalfScan6** (http://www.habets.pp.se/synscan/programs.php?prog=halfscan6) – Open source port scanner.
- **Strobe** (http://www.tuxfinder.com/packages/?defaultname=strobe&nodesc=1) -- Open source port scanner.
- **Snort** (http://www.snort.org) – An open source Intrusion Dectection System (IDS) with limited and very experimental IPv6 support.
- **ISS RealSecure 7.0 and Proventia** (http://www.iss.net) – Commercial Intrusion Detection Systems (IDS) with IPv6 support.
- **NFR Sentivist 4.0** (http://www.nfr.com) -- Commercial Intrusion Detection Systems (IDS) with IPv6 support.
- **Ethereal** (http://www.ethereal.com) – Open source packet sniffer and analyzer with full IPv6 support.
- **NetCat6** (http://netcat6.sourceforge.net) -- Netcat6 is a simple Unix utility which

reads and writes data across IPv6 or IPv4 network connections. It is designed to be a reliable "back-end" tool that can be used directly or easily driven by other programs and scripts.

- **mPing** (http://www.cdt.luth.se/~nord/progs/mPing) -- mPing is a tool for monitoring the multicast status within a network. Full IPv6 support.

- **TCPDump**(http://www.tcpdump.org) – Open source program that dumps traffic on a network. Full IPv6 support. There are several other programs (COTS and F/OSS) with IPv6 support that have roughly the same functionality, including Solaris Snoop, COLD, Analyzer, WinDump, WinPCAP, NetPeek, and SnifferPro.

- **SendIP** (http://www.earth.li/projectpurple/progs/sendip.html) -- SendIP is an open source command line tool to allow sending arbitrary IP packets. Full IPv6 support.

While this is no means a fully comprehensive list, the number of IPv6 tools does not hold a candle to the number of security based applications available for IPv4. A good list of these applications is available at http://www.insecure.org/tools.html.

In a global IPv6 environment, we actually encounter several of the same security issues that we also encounter in an IPv4 world. IPv6 is only one layer of the OSI model, we still have 6 other layers where security is a major concern. A good example of this appeared recently in a major security issue found in the TCP protocol layer ( http://www.uniras.gov.uk/vuls/2004/236929/index.htm ). This is a layer 4 problem, it really has nothing to do with the underlying protocol that runs TCP (in this case IPv4 or IPv6). Both IPv4 and IPv6 use BGP for routing, therefor both protocols are vulnerable. One of the advantages of IPv6 in this case, however, is that with an exponentially larger address space such as that provided with IPv6, the chances of having easy to guess address end points dwindles astronomically.

Along the lines of more addresses, while hiding in the 340 undecillion addresses of IPv6 may be considered "security through obscurity" by most security experts, we must take something into consideration. Blaster/Slammer and other "network scanning as part of the attack/infection vector" worms and viruses would not be nearly as big a problem in a purely IPv6 environment. With current scanning techniques, it is possible, with the right equipment and software, to scan the entire IPv4 internet in 10 hours (from http://www.opte.org/history/ ). Based on that calculation:

**2^32 addresses / 429,496,729 IP addresses per hour = ~10 hours**

**2^128 addresses / 429,496,729 IP addresses per hour = ~90.44 * 10^24 years**

These figures completely ignore the fact that many of these IPv4 addresses are not routable, or are simply not used, and the "internet scanning" tools most likely are going to take this into account for both the IPv4 and IPv6 address space. However, I suspect the time lines are not going to differ dramatically because of this (the ratio of number of scannable IPs to number of IPs scanned per hour will stay roughly the same).

Another advantage of IPv6 ties into the "crunchy outer shell and chewy center" network analogy:

Many SOHO, home user and small to medium businesses use NAT (Network Address Translation) as the PRIMARY (and many times ONLY) security for their network. NAT was not designed as a security mechanism, it was designed as a patch for a shortage of IPv4 addresses.

A stateful inspection firewall, is, on the other hand, a very good security mechanism. It is NOT, however, the end-all-be-all for security. There is no silver bullet for security. Network security is to be implemented in layers. Part of this layering is actually locking down and patching the devices that are behind said firewall. IPv6 is not going to change the requirement of patching systems. Vulnerabilities will be found and either reported by the white hat community or exploited by the black hat community.

A benefit derived from a global deployment of IPv6 is that NAT is no longer required. Machines that at one point were thought to be safe, simply because they were behind a NAT server, will no longer have that false sense of security. True stateful firewalls will have to be implemented and hardened, and the servers and workstations behind those firewalls will also have to be hardened. This will strengthen overall security in a global Internet environment.

One of the strengths of IPv6 could also be considered one if its major weaknesses. IPv6 includes the capability for end-to-end encryption and IPSec, ensuring both confidentiality and integrity of data passed between two hosts. The major problem with this method of encryption is intrusion detection and intrusion prevention systems on the borders of networks. If end-to-end encryption is in place between hosts on two different networks, an IDS (Intrusion Detection System) or IPS (Intrusion Prevention System) will only see encrypted packets, making it almost impossible to discover if the traffic is benign or malicious.

A fix for this is the IDS or IPS having all the cryptographic keys for all the clients on the network stored locally. This causes yet another security issue in the event the IDS or IPS is compromised. Now an attacker has every cryptographic key for the target network. This fix also does not address the fact that a network's IDS or IPS is going to require a LOT more processing power, as it has to decrypt and possibly re-encrypt every encrypted packet going through it. While this is possible in a distributed networking environment, many small to medium businesses simply do not have the capital to implement this type of infrastructure for their networks.

Many of the security issues that are occurring and will occur in IPv6 tie into transitioning mechanisms from IPv4. While this is not an inherent security issue with IPv6 itself, it is something that must be addressed, as a full transition to the IPv6 protocol is most likely many years down the road. In the meantime, the addressing schemas will have to be able to communicate with each other; so transitioning and tunneling mechanisms are required to accomplish this goal. There are several such methods available. The choice of method(s) will vary from business to business and application to application, based on the requirements of each network, administrator experience, and

network topology. A follow on paper for the implemented security mechanisms at this time is future work in progress and currently being worked on to be delivered in the near future via the NAv6TF.

This is by no means a comprehensive list of the security issues or possibilities arising from both native IPv6 or IPv6 transitions, it is meant to give an idea of the seriousness of the issues at hand. Many of these attacks are also possible in the IPv4 only world (MITM, DoS and DDoS). The services that unknowingly assist in these attacks are available with both protocols (HTTP, FTP, HTTPS, SSH, IRC, etc.). The important issue is education. Before a business rolls out IPv6, an administrator should know exactly what they are getting in to, either via self study or third party training. IPv6 is a very powerful protocol, but both white hats and black hats understand its strengths and weaknesses. Learning the protocol before rolling it out, or at the very least testing it in a lab environment before roll out will benefit everyone in the transition to IPv6.